

Cosas a realizar:

1. Leer detenidamente la descripción del sistema y de lo que se va a mostrar.
 - a. Por ejemplo, las facturas del sistema deben aparecer en la interfaz del cliente.
 2. Siguiendo las pautas de la diapositivas, definir un verbo que permita realizar la acción o acciones requeridas para realizar lo que se pide.
 - a. En este caso, necesitamos recuperar las facturas. Así que definiremos un verbo GET que recupere facturas.
 3. Tras eso, convendría probar los verbos una vez se implementen en el sistema consultando su URL a través de la web.
- Los aggregate roots se ponen como el nombre en plural

La URL será `http://polaflix_nelson.nelson.unican.es<Sustituir por la IP local>/facturas/{id}/items`

Recurso: `/facturas/{id}/items`

Verbos Aceptados: GET, POST

Respuestas: 200 Ok, 400 Bad Request

GET `/facturas/{id}/items` - Devuelve todos los items de una factura.

Parámetros:

<Ninguno>

Modelo de Respuesta:

```
[
  {
    {
      "id" = 0,
      "fecha" = 08/07/2002,
      "precio" = 20.4,
    },
    {
      ...
    },
    ...
  ]
```

]

POST /facturas/{id}/items - Devuelve todos los items de una factura.

Parámetros:

Serie – Nombre de la serie a visualizar

Temporada – Número de la temporada

Capitulo – Número de capítulo

Han de estar especificados los tres parámetros.

Respuestas: 200 Ok, 400 Bad Request

Usuario

Recurso: /usuarios/{id} → esto devuelve todo (todos los campos del usuario)

Verbos Aceptados: GET, POST

Respuestas: 200 Ok, 400 Bad Request

GET /usuarios/{id} - Devuelve todos campos de un usuario:

<Ninguno>

Modelo de Respuesta:

```
[
  {
    {
      "id" = 0,
      "tarifaPlana" = true,
      "empezadas" = {
        {
          "id" = 2,
          "titulo" = "breaking bad" ,
          "temporadas" =
            {
              {
```

```

        "id" = 1,
        "numeroTemporada" = 1,
    },
    },
    },
    {
        ...
    },
    ...
}

```

POST /usuarios/{id} -

 nombreUsuario – Nombre del usuario

 contrasenia – Contraseña de la cuenta

Han de estar especificados los dos parámetros.

Respuestas: 200 Ok, 400 Bad Request

Visualizacion

Recurso: /visualizaciones/{id}

Verbos Aceptados: GET, POST

Respuestas: 200 Ok, 400 Bad Request

GET /visualizaciones/{id} - Devuelve todas las visualizaciones con ese id

Parámetros:

 <Ninguno>

Modelo de Respuesta:

```
[
  {
    "id" = 0,
    "fecha" = 08/07/2002,
    "precio" = 20.4,
    "Usuario" = {
      "id" = 0,
      "nombreUsuario" = "pepe",
      ...
    }
  }
]
```

POST /visualizaciones/{id} - Devuelve todas las visualizaciones con ese id

Parámetros:

fecha – fecha de la visualización

precio – precio de la visualización

usuario – la cuenta del usuasrio

Han de estar especificados los tres parámetros.

Respuestas: 200 Ok, 400 Bad Request

Serie

Recurso: /series/{id}

Verbos Aceptados: GET, POST

Respuestas: 200 Ok, 400 Bad Request

GET /series/{id} - Devuelve todos los campos de la serie con ese id.

Parámetros:

<Ninguno>

Modelo de Respuesta:

```
[
  {
    "id" = 0,
    "titulo" = "Pokémon",
    "precio" = 20.4,
    "temporadas" = {
      {
        Id = 0,
        ...
      },
      ...
    }
  }
]
```

POST /series/{id} - Devuelve todos los campos de la serie con ese id.

Parámetros:

titulo – Nombre de la serie a visualizar

Temporada – id de la temporada

Han de estar especificados los dos parámetros.

Respuestas: 200 Ok, 400 Bad Request